Génération d'hypothèses de façades utilisant des critères contextuels et structurels

Generation of facade hypotheses based on contextual and structural information

Antoine Fond Marie-Odile Berger Gilles Simon

LORIA, Université de Lorraine, INRIA

LORIA, 615 Rue du Jardin botanique, 54506 Vandoeuvre-lès-Nancy
antoine.fond@inria.fr

Résumé

Dans cet article nous nous intéressons à la détection de façades dans le but d'améliorer la mise en correspondance image/modèle de bâtiments pour le calcul de pose en milieux urbain. Après une étape de rectification automatique, nous employons un schéma en deux étapes. Premièrement une cascade de classifieurs LogitBoost basés sur des indices simples faisant intervenir le contexte local sélectionne quelques fenêtres parmi un ensemble de fenêtres tirées selon une distribution a priori. Ces façades potentielles sont ensuite décrites plus structurellement par leur représentation de Haar-Fourier. Elles sont finalement retenues ou écartées par un classifieur fort SVM. Les résultats sont évalués sur une base de test de 410 images urbaines.

Mots Clef

Détection d'objets, façade, objectness, classification

Abstract

In that article we focus on facade detection in order to improve image/model buildings matching for pose computation in urbain environnement. We use a two-step design. First a cascade of LogitBoost classifiers using features which describe local context selects a few windows from a set of windows drawn on an a priori distribution. These facade candidates are then more internally described using their Haar-Fourier representation. Eventually they are discarded or kept by a strong classifier SVM. Results are computed from a 410-set of urban images.

Keywords

Object detection, façade, objectness, classification

1 Introduction

Si la réalité augmentée en environnement urbain offre de belles perspectives d'application, elle se heurte à des difficultés techniques qui peuvent être surprenantes. On pourrait penser que dans un tel environnement modelé par l'homme le calcul de pose de caméra soit plus simple. En réalité, les méthodes classiques à base de points d'intérêts échouent souvent à résoudre cette étape clé dans l'introduction d'objets virtuels dans la scène. En effet la présence de nombreux motifs répétés constitutifs des façades est à l'origine de biais en translation. Les dimensions relatives des bâtiments, des piétons et des rues sont à l'origine d'effets de perspective souvent importants dans les images issues de ce type d'environnement. C'est une difficulté supplémentaire pour les méthodes qui reposent sur des descripteurs dont l'invariance au changement de point de vue est limitée.

Nous avons proposé dans un précédent article une méthode rapide pour rectifier des images en environnement urbain [12]. Nous nous intéressons à présent plus spécifiquement à la mise en correspondance de bâtiments dans de tels environnements. Si la rectification permet de s'abstraire des déformations projectives importantes, il reste difficile dans de telles images de limiter la zone de correspondances. Nous proposons ici une méthode rapide de détection de façade dans des images préalablement rectifiées. Cela permet de limiter les correspondances dans le problème de calcul de pose mais aussi de limiter les candidats pour la reconnaissance de bâtiments. Ce problème est à distinguer du «facade parsing». On ne cherche pas ici à segmenter sémantiquement une façade en sous-éléments mais bien à détecter des candidats façades dans une image urbaine qui en contient. Par conséquent le modèle de façade utilisé ici repose sur des critères structurels simples mais aussi sur le contexte local.

La première contribution réside dans la définition d'indices adaptés à la modélisation de façade en environnements urbains. Ces indices sont utilisées dans un classifieur en cascade pour limiter le nombre de candidats qui doivent passer par un second classifieur fort. La définition d'un descripteur adapté à la classification de façade basé sur la transformée en ondelettes de Haar et la transformée de

Fourier constitue notre seconde contribution. Nous avons également construit une base de 920 images d'environnements urbains issues de sous-ensembles «street» et «building» d'«ImageNet»[10]. Toutes les images ont été rectifiées par rapport à la façade prédominante et nous avons détouré manuellement les façades pour un total de 1324 façades sur l'ensemble de la base. 510 images sont dédiées à l'apprentissage des paramètres de la méthode et 410 sont utilisées pour le test.

1.1 Travaux antérieurs

Deux types de méthodes ont été proposées par le passé pour détecter des façades dans une image. Les premières, de nature géométrique, sont essentiellement basées sur la recherche de rectangles dont les arêtes sont cohérentes avec les points de fuite de l'image [9, 11, 5]. Les secondes, de nature statistique, utilisent diverses sortes d'indices pour identifier les régions de l'image supposées couvrir une façade [8, 2].

Dans [9], des segments de droites sont détectés automatiquement dans l'image et prolongés afin de former des hypothèses de rectangles en accord avec les points de fuite. Pour chaque hypothèse de rectangle, l'image est orthorectifiée et un histogramme de gradients (HOG) est calculé à l'intérieur du rectangle rectifié. Les hypothèses dont le HOG comporte des pics ailleurs qu'aux orientations horizontale et verticale sont rejetées. Afin d'éviter une recherche exhaustive, Mikusik et al. restreignent la détection des rectangles sur une structure de voisinage donnée par une triangulation de Delaunay [11]. Le problème est alors formulé en terme de maximisation de la probabilité a posteriori d'un champ aléatoire de Markov. Dans [5], des coins plutôt que des arêtes sont utilisés pour détecter les façades. Dans l'image rectifiée automatiquement à partir des points de fuites, les coins droits de l'image sont détectés en utilisant une machine à vecteurs de support (SVM). Une triangulation de Delaunay est ensuite effectuée sur les coins droits et un algorithme de type min-cut est utilisé pour déterminer des régions rectangulaires à l'intérieur desquelles une concentration importante de coins droits est observée. Ces différentes méthodes permettent d'obtenir des structures rectangulaires présentes sur les façades (fenêtres, groupes de fenêtres...) mais malheureusement pas (en général) les façades entières.

D'autres méthodes visent à segmenter l'image en régions associées à différents labels, dont un label associé aux façades. Ces méthodes utilisent des indices de natures différentes au sein d'algorithmes de classification supervisée. Hoiem et al. ont ouvert la voie à ces travaux, en permettant d'associer à des régions de l'image (des constellations de superpixels) les catégories "sol", "ciel" et "vertical" [8]. Une version d'Adaboost utilisant la régression logistique est employée pour l'attribution des labels. Les classifieurs faibles reposent sur une trentaine de caractéristiques mesurées à l'intérieur des superpixels et des constellations, incluant des mesures de couleur, de texture, de position, de forme et de géométrie. Dans ces travaux, le label "vertical" peut être attribué à n'importe quel objet vertical. Une extraction plus spécifique de façades est proposée dans [2] dans le cadre d'un calcul de pose. Des patchs sont extraits autour de chaque pixel de l'image et caractérisés par des indices ICF (*Integral Channel Features*). Une SVM est utilisée pour classifier les patchs entre "façade", "ciel", "toit", "végétation" et "sol". Ce type d'approche suppose que des patchs locaux contiennent suffisamment d'information pour permettre de décider de leur appartenance à l'une des classes. Un grain de segmentation plus gros doit cependant être considéré pour tirer profit d'indices plus globaux caractéristiques d'une façade. Plus récemment Farabet et al. ont proposé dans [4] une approche multi-résolution par réseau de neurones convolutionels à ce problème de segmentation sémantique de scène. Si les pixels correspondant aux façades sont bien classifiés, la géométrie des façades est ignorée et aucune distinction n'est possible entre les bâtiments. Cela se montre limitant si l'objectif final est la reconnaissance de ces bâtiments ou calcul de pose à partir de ceux-ci.

La méthode présentée dans cet article est à la fois de nature géométrique (nous considérons des images rectifiées et cherchons à extraire des rectangles dont les côtés correspondent aux bords des façades) et de nature statistique (un apprentissage supervisé basé sur une combinaison de divers indices est utilisé pour identifier de tels rectangles). La détection des façades repose sur un schéma en deux étapes, dans l'esprit de [7, 14], avec un premier classifieur rapide permettant d'extraire un nombre limité de fenêtres candidates, suivi par un classifieur structurel plus élaboré. Nous nous inspirons des travaux présentés dans [1], qui caractérisent un objet dans son environnement à l'aide de plusieurs indices (contours connexes, contraste de couleur,...) calculés sur un ensemble de fenêtres parcourant l'image. Un classifieur bayésien naif combinant ces indices permet ainsi de sélectionner un nombre limité de candidats objets. Dans l'esprit de la mesure d'«objectness» proposée par Alexe et al., nous définissons dans cet article une mesure de «facadeness». Cependant, si certains indices utilisés par Alexe et al. ont pu être repris, d'autres sont inapplicables dans notre contexte. En effet, un de leurs indices majeurs repose sur une certaine distribution fréquentielle des images que l'on observe pas en environnement urbain.

Nous présentons en section 2 l'étape de sélection des fenêtres candidates basée sur les indices de «facadeness». Afin de limiter la combinatoire nous n'utilisons pas une méthode de fenêtres glissantes mais générons des candidats à partir d'une distribution apprise sur une base d'exemples. La deuxième étape, présentée en section 3, utilise un classifieur de type SVM basé sur un descripteur structurel des façades de type Haar-Fourier. Notre méthode est évaluée quantitativement en section 4 grâce à une base annotée manuellement. Plusieurs exemples de façades obtenues par la méthode sont également montrés.

2 Recherche de fenêtres candidates

2.1 Distribution *a priori* de fenêtres

La plupart des méthodes de détection d'objets dans une image nécessitent de parcourir celle-ci pour un ensemble de fenêtres. L'ensemble le plus évident est une discrétisation de l'espace 4D (x, y, l, h) selon une grille régulière. Dans notre cas si la forme rectangulaire des fenêtres est bien adaptée à la recherche de façades rectifiées, certaines mesures de sélection des candidats utilisent les bords de la façade et nécessitent donc un bon ajustement de la fenêtre. Une telle précision se traduit par une grille de parcours plus fine. Aussi bien pour l'apprentissage que pour les tests, nous utilisons le critère suivant $s_{IoU} = \frac{w_d \cap w_{gt}}{w_d \cup w_{gt}}$ pour mesurer l'ajustement d'une fenêtre parcourue w_d par rapport à une fenêtre de la vérité terrain w_{gt}. Ainsi pour pouvoir espérer trouver parmi les fenêtres parcourues l'ensemble des fenêtres de la vérité terrain avec un score $s_{IoU} \geq 0.8$ il faut déjà une grille régulière de l'ordre de 10^7 fenêtres. En fait seule une infime partie de cet ensemble à des chances d'être un candidat façade. En effet les contraintes architecturales et de prises de vue restreignent la forme de fenêtre (*aspect ratio*) et sa position dans l'image. Il est par exemple très peu probable de trouver une façade très allongée horizontalement dans le coin inférieur droit de l'image.

Notre stratégie est d'apprendre cette loi de probabilité $p(W), W \in \mathbb{R}^4$ sur un ensemble d'exemples et de tirer aléatoirement les fenêtres selon celle-ci. On choisit d'utiliser un estimateur à noyau gaussien pour approcher $p(W)$. La variance des gaussiennes est choisie de manière à minimiser le nombre de tirages pour avoir 90% de chances d'avoir une fenêtre correcte ($s_{IoU} \geq 0.8$) parmi celles du tirage et ceci pour toutes les images de la base d'apprentissage. Pour cela on fait varier la variance $\Sigma^2 = diag(\sigma_x^2, \sigma_y^2, \sigma_h^2, \sigma_l^2)$. Pour chaque valeur du paramètre Σ^2 on estime la distribution $p(W)$. Pour chaque image on tire alors successivement $(10, 10^2, ...10^7)$ exemples. Pour chacun de ces tirages on calcule le nombre de fenêtres correctes en ne comptant au maximum qu'une fenêtre correcte par fenêtre de la vérité terrain. On cherche alors le tirage dont le taux total de fenêtres correctes sur le nombre total de fenêtres de la vérité terrain est supérieur à 90%.

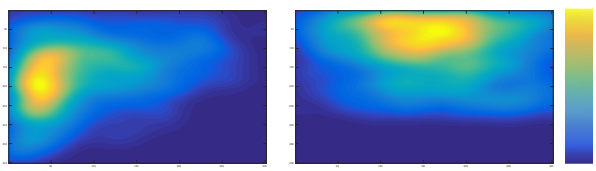

FIGURE 1 – Marginales de la distribution *a priori* $p(W)$ en (h, l) et en (x, y) respectivement à gauche et à droite.

2.2 Indices de «facadeness»

On peut limiter raisonnablement le nombre de fenêtres à tester à 100000 en tirant profit de $p(W)$. En effet cela

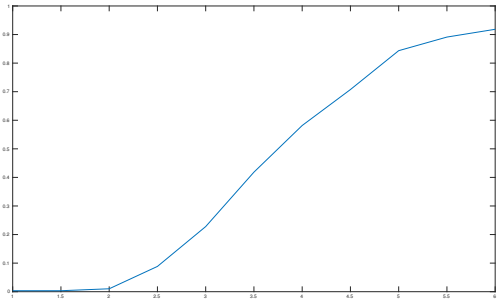

FIGURE 2 – Taux total de fenêtres correctes sur l'ensemble de la base d'apprentissage en fonction du nombre de tirages en échelle logarithmique.

représente un taux de fenêtres correctes de l'ordre de 80% sur la base d'apprentissage comme sur celle de test. Cependant ce nombre est encore bien trop important pour évaluer toutes les fenêtres par un classifieur fort dont le calcul du descripteur dépend de la taille de la fenêtre (HOG, dSIFT, ...). Certains indices simples propres aux façades peuvent être utilisés pour discriminer une grande partie de ces fenêtres et ne garder que les meilleurs candidats. Ces indices étant évalués sur un grand nombre de fenêtres ils doivent pouvoir être calculés très rapidement (dans l'idéal en temps constant par rapport à la taille de la fenêtre).

L'ensemble de ces indices constitue une première étape dans notre modélisation d'une façade dans une image. Celle-ci repose sur trois hypothèses : une façade est rectangulaire, elle est structurées par des éléments eux mêmes rectangulaires. Enfin elle se démarque de son environnement par cette même structure et par sa couleur. On introduit la notation $w = W(x, y, l, h)$ le rectangle à la position (x, y) de largeur l et de hauteur h. On notera également $r = R(I, w)$ la région de l'image I comprise dans w.

Une façade dans une image rectifiée est rectangulaire. Elle se distingue par un contour important le long des arrêtes du rectangle w. Ainsi on définit l'indice de contour suivant :

$$\begin{aligned} s_{cp}(w) = \frac{1}{2(l+h)}(&\sum R(E_x, W(x-\alpha, y, \alpha, h)) \\ &+ \sum R(E_x, W(x+l+\alpha, y, \alpha, h)) \\ &+ \sum R(E_y, W(x, y-\alpha, l, \alpha)) \\ &+ \sum R(E_y, W(x, y+h+\alpha, l, \alpha))) \end{aligned} \quad (1)$$

où α représente l'épaisseur des bandes entourant w. E_x et E_y sont les images de contours respectivement horizontaux et verticaux (la figure 3 montre les valeurs de s_{cp} à toutes les positions d'une image d'exemple).
L'indice s_{cp} est calculé en temps constant à partir de l'intégrale image de l'image filtrée par un détecteur de Canny dont on ne garde que les contours orientés horizontalement

et verticalement. L'utilisation d'intégrales images pour le calcul de sommes de régions est décrit dans [15].

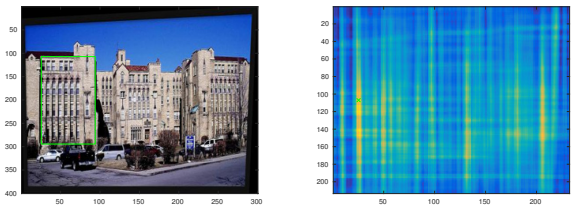

FIGURE 3 – Indice s_{cp} calculé pour toutes les positions (x, y) d'une fenêtre de taille (l, h) correspondant à la vérité terrain (en vert). La position (x, y) d'une fenêtre est celle de son coin supérieur gauche.

Une façade rectifiée est majoritairement composée d'éléments architecturaux verticaux et horizontaux (fenêtres, balcons, porte, briques ...) et de zones uniformes (peinture, crépis). Ces éléments pseudo-rectangulaires se manifestent à plusieurs échelles (le bâtiment lui même, les étages, les fenêtres, les briques,...). Ainsi sa transformée en ondelettes de Haar DWT_{haar} (figure 8) par sa dimension multi-échelle et la forme de ses ondelettes est très creuse. On mesure la structure dans les coefficients d'ondelettes par l'entropie de la décomposition. On définit alors l'indice suivant :

$$s_{he}(w) = H_S(DWT_{haar}(R(I)), w)) \tag{2}$$

où $H_S(x) = -\sum_i \frac{|x_i|}{\|x\|_1} \log(\frac{|x_i|}{\|x\|_1})$ est l'entropie de Shannon pour un vecteur x (voir la figure 4).
L'indice s_{he} est calculé rapidement à partir des intégrales images de $|x| \log(|x|)$ et $|x|$ des sous-bandes x de la transformée en ondelettes de I.

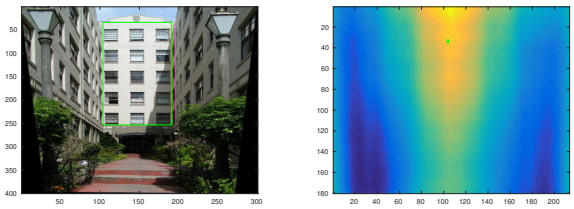

FIGURE 4 – Indice s_{he} calculé pour toutes les positions (x, y) d'une fenêtre de taille (l, h) correspondant à la vérité terrain (en vert). La position (x, y) d'une fenêtre est celle de son coin supérieur gauche.

Par ailleurs une façade d'une image I se distingue davantage suivant que son environnement proche n'est lui même pas rectifié, ou composé d'éléments à la structure complexe (arbres, buissons, ciel nuageux,...). Ainsi la distribution des coefficients d'ondelettes d'un anneau autour de la façade est très différente de celle à l'intérieur. L'entropie des coefficients à l'intérieur étant faible la distribution est piquée en zéro et sur quelques grandes valeurs. La distribution est plus plate à l'extérieur où l'inadéquation avec les atomes de Haar répartit l'énergie du signal sur un plus grand nombre de coefficients. En approximant les distributions par leurs histogrammes on peut les comparer par un test d'adéquation du χ^2. On définit alors :

$$s_{hd}(w) = \sqrt{\sum_i \frac{\left(h^H_{+\beta,i} - h^H_i\right)^2}{h^H_i}} \tag{3}$$

où $h^H_{+\beta}$ et h^H sont les histogrammes des coefficients d'ondelettes respectivement des régions de l'image $R(I, w_{+\beta})$ et $R(I, w)$ avec $w_{+\beta} = W(x - l\beta, y - h\beta, l(1+2\beta), h(1+2\beta))$ le rectangle augmenté d'un facteur β dans toutes les directions (voir la figure 5).
On discrétise l'espace des coefficients d'ondelettes en 10 baquets. En utilisant une intégrale image par baquet on calcule alors les histogrammes en temps constant (4×10 opérations).

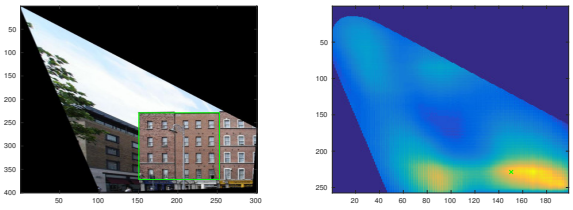

FIGURE 5 – Indice s_{hd} calculé pour toutes les positions (x, y) d'une fenêtre de taille (l, h) correspondant à la vérité terrain (en vert). La position (x, y) d'une fenêtre est celle de son coin supérieur gauche.

Enfin une façade se distingue de son environnement dans l'image par une différence dans la distribution de couleur. Ce contraste de couleur permet de discriminer une façade d'immeuble du ciel, ou une façade grise, d'une façade rouge adjacente. On utilise également un test d'adéquation du χ^2 pour comparer ces distributions. On définit alors :

$$s_{cc}(w) = \sqrt{\sum_i \frac{\left(h^C_{+\gamma,i} - h^C_i\right)^2}{h^C_i}} \tag{4}$$

où $h^C_{+\gamma}$ et h^C sont les histogrammes couleurs respectivement des régions de l'image $R(I, w_{+\gamma})$ et $R(I, w)$ avec $w_{+\gamma}$ le rectangle augmenté d'un facteur γ dans toutes les directions (voir la figure 6).
En discrétisant l'espace couleur LAB en $256 = 4 \times 8 \times 8$ baquets, on construit une intégrale image par baquet et donc calculer un histogramme en 4×256 opérations.

Les différents indices sont parametrés par α, β, γ. Ces paramètres sont appris de manière à maximiser la séparabilité des examples positifs de fenêtres par rapport aux exemples négatifs. Sur la base d'apprentissage on calcule

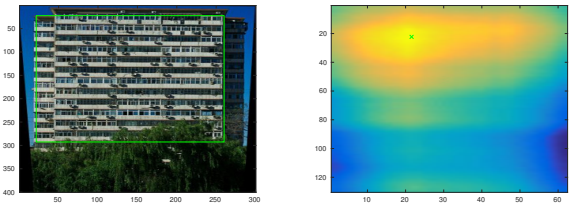

FIGURE 6 – Indice s_{cc} calculé pour toutes les positions (x, y) d'une fenêtre de taille (l, h) correspondant à la vérité terrain (en vert). La position (x, y) d'une fenêtre est celle de son coin supérieur gauche.

la probabilité conditionnelle des valeurs d'un indice sachant qu'il est calculé pour des fenêtres positives ou négatives. Cette vraisemblance d'indice est calculé par un histogramme de ses valeurs pour l'ensemble des fenêtres positives et pour l'ensemble des fenêtres négatives. On cherche alors la valeur du paramètre qui permet de séparer au maximum ces deux histogrammes. La mesure de séparabilité est le critère s_{IoU} défini précédemment pour l'aire sous les courbes. On obtient alors $\alpha = 8$, $\beta = 0.3$ et $\gamma = 0.24$ (voir la figure 7).

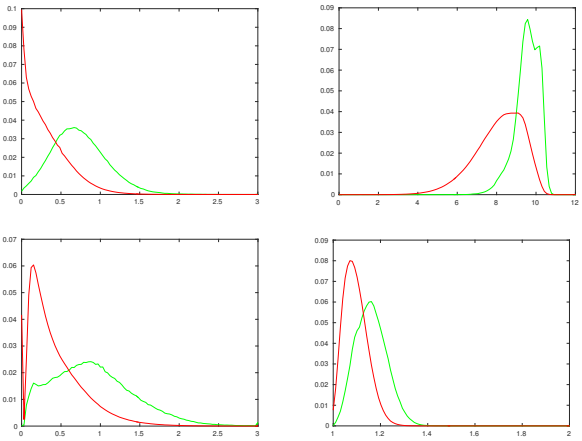

FIGURE 7 – Vraisemblance des indices s_{cp} (haut à gauche), s_{he} (haut à droite), s_{hd} (bas à gauche), s_{cc} (bas à droite). En vert la vraisemblance pour les fenêtres positives et en rouge pour les fenêtres négatives.

2.3 Cascade de classifieurs Logitboost

Si tous les indices présentés peuvent être calculés en temps constant par rapport à la taille des fenêtres, ces constantes sont plus ou moins grandes suivant les indices. Ainsi le contraste de couleur s_{cc} nécessite plus de 2^{10} opérations par fenêtre tandis que s_{cp} n'en nécessite que 2^4. Aussi on adopte un schéma de classification en cascade. L'interêt d'un tel schéma est de classifier d'abord selon les indices les moins coûteux. Les indices plus coûteux (comme s_{cc} ici) ne sont calculés que sur les survivants des étages précédents de la cascade de classifieurs. Les classifieurs utilisés pour chaque niveau de la cascade sont des classifieurs Logitboost C qui combinent linéairement N classifieurs faibles $c_{i,j}$ à seuil.

$$C(x) = \sum_{i=0}^{N} w_i c_{i,j}(x) \text{ avec } c_{i,j}(x) = \begin{cases} 1, & \text{si } x_j \geq \xi_{i,j} \\ -1, & \text{sinon} \end{cases} \quad (5)$$

La différence entre Adaboost et Logitboost provient de la fonction de coût à minimiser lors de l'apprentissage qui est alors celle d'une regression logistique [6].

La cascade est composée de 3 étages. L'apprentissage du classifieur Logitboost de chaque étage doit chercher à maximiser le taux de façades classées façades (vrais positifs) tout en ayant un taux de rejet élevé d'un étage à l'autre. En avantageant les exemples positifs par rapports aux négatifs par une certaine pondération, on force artificiellement ce comportement dans l'apprentissage des classifeurs Logitboost. Le premier étage n'utilise ainsi que s_{he} et s_{cp} avec un classifieur Logitboost à 16 classifieurs faibles. Le second étage utilise s_{he},s_{cp} et s_{hd} avec un classifieur à 32 classifieurs faibles et enfin le dernier étage combine tous les indices s_{he}, s_{cp}, s_{hs} et s_{cc} avec 64 classifieurs faibles. Le résultat d'une telle cascade sur notre base d'apprentissage de l'ordre de 90% de rejet des 100000 tirages initiaux selon $p(W)$ et 85% de vrais positifs.

Sur les 10000 fenêtres qui restent classées positives à l'issue de la cascade beaucoup se recouvrent fortement. La non uniformité de $p(W)$ renforce cet effet. Une dernière étape consiste à ne sélectionner qu'un sous ensemble de ces fenêtres survivantes qui ont un bon score de classification par la cascade tout en ne se recouvrant pas. On utilise la même approche gloutonne que pour l'«objectness» [1] qui consiste à prendre la fenêtre de meilleur score puis à chercher la fenêtre de meilleur score qui ne la recouvre pas et ainsi de suite. Le score de classification de la «facadeness» n'étant pas assez fiable pour une approche gloutonne, on autorise en fait un recouvrement jusqu'à 50 %. On garde ainsi en pratique de l'ordre de 200 candidats façades.

3 Classification de façade

3.1 Descripteur de Haar-Fourier

La modélisation faite d'une façade par l'étape de sélection précédente est assez sommaire. On propose ici une modélisation plus fine qui permet d'extraire des caractéristiques structurelles de façades. Comme on l'a présenté précédemment on considère ici que les façades sont composées d'éléments rectangulaires d'échelles différentes. Ces éléments sont souvent répétés selon la direction horizontale ou verticale de la façade. Ces deux observations nous ont amené à construire un descripteur de façade formé de la transformée de Fourier $\mathcal{F}$ du module de chaque sous-bande (d'échelle j et de direction $d = 1$ horizontale, $d = 2$ verticale ou $d = 3$ diagonale) d'une transformée en ondelettes de Haar.

$$D = \left(\left|\mathcal{F}\left(\left|I \star W_{j,d}\right|\right)\right|^2\right)_{j \leq J, d \leq 3} \quad (6)$$

Les répétitions présentes à différentes échelles sur différentes zones de la façade (étages, fenêtres,...) rendent difficiles l'exploitation de sa transformée de Fourier globale. La transformée en ondelettes permet de décomposer le signal en plusieurs échelles et directions. Une telle décomposition rend l'analyse fréquentielle plus ciblée et plus robuste à des fréquences parasites (obstructions, parties non rectifées,...). Cependant, la transformée a tendance à étaler le spectre à chaque niveau d'échelle ou de direction, ce qui le rend plus sensible à de petites perturbations (fenêtre décalée par exemple). En effet, on peut considérer le cas d'une façade idéale sous forme de train d'impulsions. La transformée en ondelettes tend alors à le transformer en peigne de Dirac. Afin de reconcentrer le spectre vers les basses-fréquences, on prend ainsi le module de la transformée en ondelettes comme cela est fait pour le scattering d'ondelettes [3].

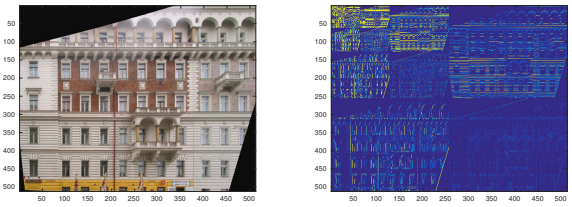

FIGURE 8 – Transformée en ondelettes de Haar d'une image de façade

En pratique si des répétitions ont lieu dans une direction donnée, l'énergie du spectre est concentrée autour de cette direction (figure 9). Afin de réduire la taille du descripteur D on supprime les coefficients hautes fréquences des diagonales. Le descripteur est calculé en redimensionnant la taille des fenêtres à 128×128. Seuls les coefficients compris dans les carrés inscrits de chaque spectre de sous-bande sont gardés. Le vecteur D est ainsi de dimension 8192. Une étape de mutualisation des coefficients voisins par maximum («maxpooling») [10] réduit encore le nombre de dimension à 1280. La taille des voisinages est divisée par deux d'une échelle à l'autre en partant d'une taille de 4×4 pour la plus haute résolution ($J = 3$).

3.2 Classification SVM

Nous exploitons cette description structurelle d'une façade comme dernière étape de la détection de façade. La classification binaire «façade»/«non-façade» utilise un classifieur linéaire à vecteurs supports SVM sur les descripteurs $D(w)$ des fenêtres issues de la cascade. La description faite par D d'une façade ne fait pas intervenir le contexte mais seulement l'intérieur de la façade. Aussi plutôt que d'utiliser la vérité terrain de notre base contextuelle limité pour l'apprentissage du SVM nous choisissons de constituer une seconde base d'image de façade seules.

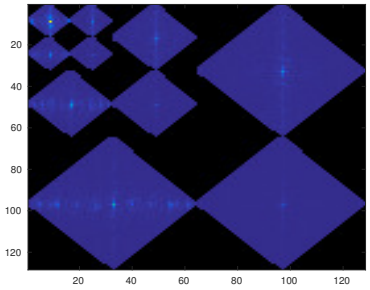

FIGURE 9 – Visualisation du descripteur de Haar-Fourier

Celle-ci est la concaténation de bases publiques utilisées pour la segmentation et totalise 2361 façades étiquetées positives [13]. Les examples négatifs sont issues d'images de contexte urbain provenant d'«ImageNet» (arbres, rue, voitures, signalisation, ciel, façades non rectifiées ...) pour un total de 5540 images. Cette base est divisée en 1574 images d'examples positifs «façade» et 3650 images d'examples négatifs «non-façade» pour l'apprentissage le reste étant dédié au test. Le taux de bonne classification est de 98 % sur la base d'apprentissage et de 95 % sur la base de tests.

4 Résultats

À la sortie de l'étape de «facadeness» on récupère en moyenne 220 fenêtres qui sont potentiellement des façades (avec un écart type de 93). Pour évaluer cette étape on compte le nombre de fenêtres correctes parmi ces candidats façades au sens de s_{IoU} avec seulement une fenêtre correcte par fenêtre de la vérité terrain. On fait cela pour chaque image de la base d'apprentissage. Le taux de fenêtres correctes en sortie de cascade est alors de 69.6 % sur l'ensemble de la base. Ce résultat est à mettre en perspective au taux de façades correctes de 80.3 % à l'entrée de cascade. Autrement dit, sur les 100000 candidats initiaux on à 80 % de chance de retrouver la vérité terrain. Sur les 200 candidats retenus parmi ces tirages initiaux cette probabilité est presque de 70 %. Ces résultats se généralisent bien sur la base de test avec des taux de façade correctes très voisins de 80.1 % avant cascade et de 68.2 % après.

La modélisation d'une façade faite par la «facadeness» ne suffit pas à définir une façade. Dans les fenêtres retenues en fin de cascade on retrouve de larges zones uniformes bordées de détails plus complexes. Celles-ci peuvent correspondre par exemples à une zone de ciel entre des immeubles, ou une enclave de routes entres des voitures. Dans ces cas tous les indices de la «facadeness» peuvent être élevés sans pour autant représenter une façade. Une modélisation plus structurelle est nécessaire pour écarter ces cas. C'est le rôle de la classification SVM sur les descripteurs de Haar-Fourier. Après cette étape le nombre de fenêtres est réduit à 110 en moyenne avec un écart type de 75. Le taux de façades correctes passe à 66 % sur la base d'apprentissage et 65 % sur la base de test. La division par deux

étapes	0	1	2	3	4	5	6
n	10^7	10^5	5.1×10^4	3.5×10^4	1.5×10^4	220	115
τ	1	0.80	0.79	0.76	0.74	0.70	0.65

FIGURE 10 – n nombre moyen de fenêtre à la fin de l'étape i. τ le taux de façade correctes parmi les n fenêtres de l'étape i. Etape de référence 0 : grille régulière ; Etape 1 : tirage selon $p(W)$; Etape 2 :1er étage de la cascade ; Etape 3 : 2e étage de la cascade ; Etape 4 : 3e étage de la cascade ; Etape 5 : non-recouvrement ; Etape 6 : classification SVM

seulement du nombre de fenêtres s'explique par le nombre important de sous parties de façades se recouvrant (l'étape de non recouvrement autorisant $s_{IoU} \geq 0.5$) qui sont donc classées elles-même «façade» par la SVM. En effet sans contexte on ne peut distinguer une sous-partie d'une façade d'une façade complète. Si on supprime un étage d'une façade cela reste une façade.

Aussi pour visualiser les résultats on construit une carte qui représente le chevauchement des sous-parties de façades (figure 11). Pour un pixel de l'image on fait la somme pondérée par le score de «facadeness» des fenêtres qui le recouvrent. Ainsi cette carte représente la densité de candidats façade à la fin de notre méthode. Comme il est difficile de représenter 100 fenêtres qui se chevauchent sur une même image, on ne choisit de montrer que les 10 premières classées selon le score de la carte intégré sur chacune des fenêtres.

Si le chiffre de 65% peut sembler faible il est à mettre en regard avec la capacité qu'a un humain à sélectionner une façade dans une image. Pour trois sélections manuelles de la base d'apprentissage faites par des personnes différentes on obtient des taux de recouvrement entre les ensembles de vérité terrain de chacun de l'ordre de 40% seulement. Ce n'est pas tant le score de recouvrement $s_{IoU} \geq 0.8$ qui est responsable de cette variabilité mais la difficulté à choisir de segmenter un même bâtiment en une ou plusieurs façade, et à les sélectionner de manière plus ou moins exhaustive dans l'image.

Le temps moyen de la «facadeness» est de 2.2s par image pour un code Matlab exécuté sur un Intel Xeon W3565 Quadcore 3.2 Ghz avec 64Go RAM. Le calcul rapide des indices sur les fenêtres est codé en C et seulement appelé par Matlab. Le temps moyen passe à 4.5s avec l'étape de classification SVM.

5 Conclusion

Nous avons présenté une méthode rapide de détection de façades dans la perspective d'isoler des régions d'intérêt pour la mise en correspondance modèle / image. Le nombre d'hypothèses finalement retenues, de l'ordre de 100, peut paraître encore relativement élevé. Il correspond toutefois à un petit nombre de zones d'intérêt, qui sont en bon accord avec les façades présentes dans l'image et pourront être exploitées pour l'appariement et le calcul de pose.

Nous souhaitons par ailleurs poursuivre l'investigation du critère Fourier-Haar et déterminer dans quelle mesure ce critère peut être utilisé comme descripteur global pour l'indexation de façade.

Références

[1] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11) :2189–2202, 2012.

[2] C. Arth, C. Pirchheim, J. Ventura, D. Schmalstieg, and V. Lepetit. Instant outdoor localization and SLAM initialization from 2.5D maps. *IEEE Transactions on Visualization and Computer Graphics*, 21(11) :1309–1318, 2015.

[3] J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8) :1872–1886, 2013.

[4] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.

[5] A. Fond, M.-O. Berger, and G. Simon. Prior-based facade rectification for AR in urban environment. In *ISMAR workshop on Urban Augmented Reality*, Fukuoka, Japan, 2015.

[6] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression : a statistical view of boosting. *Annals of Statistics*, 28 :2000, 1998.

[7] H. Harzallah, F. Jurie, and C. Schmid. Combining efficient object localization and image classification. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 237–244, 2009.

[8] D. Hoiem, A. A. Efros, and M. Hebert. Automatic photo pop-up. In *ACM SIGGRAPH 2005 Papers*, pages 577–584, NY, USA, 2005.

[9] J. Košecká and W. Zhang. Extraction, matching, and pose recovery based on dominant rectangular structures. *Computer Vision and Image Understanding*, 100(3) :274 – 293, 2005.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.

[11] B. Mičušík, H. Wildenauer, and J. Košecká. Detection and matching of rectilinear structures. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2008.

[12] G. Simon, A. Fond, and M.-O. Berger. A Simple and Effective Method to Detect Orthogonal Vanishing Points in Uncalibrated Images of Man-Made Environments. In *Eurographics 2016*, Lisbon, Portugal, 2016.

[13] R. Tyleček and R. Šára. Spatial pattern templates for recognition of objects with regular structure. In *Proc. GCPR*, Saarbrucken, Germany, 2013.

[14] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009.

[15] P. Viola and M. Jones. Robust real-time object detection. In *International Journal of Computer Vision*, 2001.

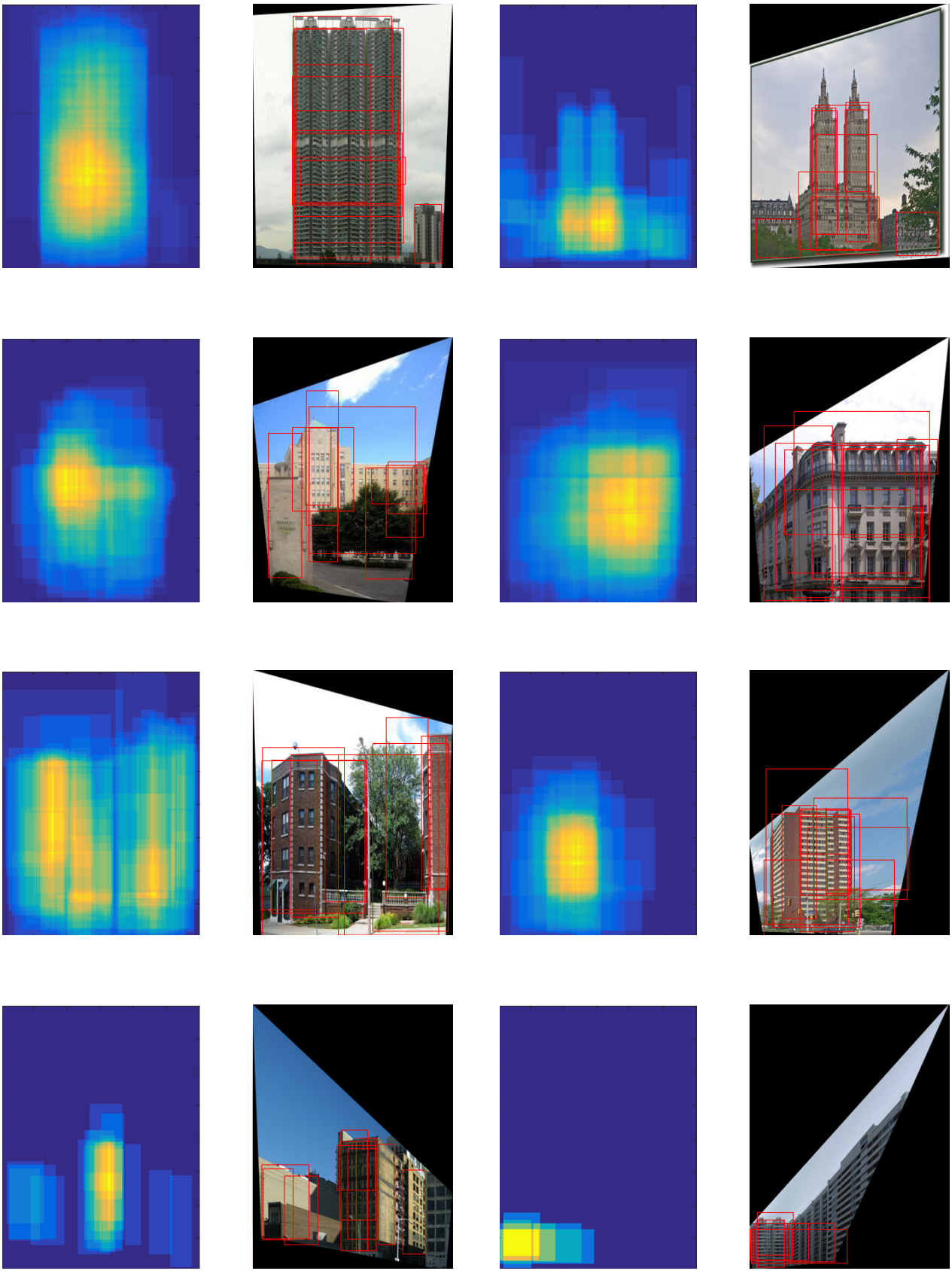

FIGURE 11 – Somme pondérée par le score de «facadeness» des fenêtres retenues par le SVM et 10 premières fenêtres classées selon le score de cette carte intégré à l'intérieur de chacune des fenêtres.